Identifying User Profile by Incorporating Self-Attention Mechanism Based on CSDN Data Set

Junru Lu[1], Le Chen[1], Kongming Meng[1,2], Fengyi Wang[3], Jun Xiang[1], Nuo Chen[1], Xu Han[4] & Binyang Li[1†]

[1]School of Information Science and Technology, University of International Relations, Beijing 100091, China

[2]Deep Brain Co. Ltd., Shanghai 201200, China

[3]University of Chinese Academy of Sciences, Beijing 100049, China

[4]College of Information Engineering, Capital Normal University, Beijing 100048, China

ABSTRACT

With the popularity of social media, there has been an increasing interest in user profiling and its applications nowadays. This paper presents our system named UIR-SIST for User Profiling Technology Evaluation Campaign in SMP CUP 2017. UIR-SIST aims to complete three tasks, including keywords extraction from blogs, user interests labeling and user growth value prediction. To this end, we first extract keywords from a user's blog, including the blog itself, blogs on the same topic and other blogs published by the same user. Then a unified neural network model is constructed based on a convolutional neural network (CNN) for user interests tagging. Finally, we adopt a stacking model for predicting user growth value. We eventually receive the sixth place with evaluation scores of 0.563, 0.378 and 0.751 on the three tasks, respectively.

1. INTRODUCTION

Social media have recently become an important platform that enables its users to communicate and spread information. User-generated content (UGC) has been used for a wide range of applications, including user profiling. The Chinese Software Developer Network (CSDN) is one of the biggest platforms of software

---

† Corresponding author: Binyang Li (Email: byli@uir.edu.cn; ORCID: 0000-0001-9013-1386).

developers in China to share technical information and engineering experiences. Analyzing UGC on the CSDN can uncover users' interests in the software development process, such as their past interests and current focus, even if their user profiles are incomplete or even missing. Apart from the UGC, user behavior data also contain useful information for user profiling, such as "*following*," "*replying*," and "*sending private messages*," through which the friendship network is constructed to indicate user gender [1,2,3], age [4], political polarity [5, 6] or profession [7].

In SMP CUP 2017 [8], the competition is structured around three tasks based on CSDN blogs[①]: (1) keywords extraction from blogs, (2) user interests labeling and (3) user growth value prediction. Our team from School of Information Science and Technology, University of International Relations participated in all the tasks in User Profiling Technology Evaluation Champaign. This paper describes the framework of our system UIR-SIST for the competition. We first extract keywords from a user's blog, including the blog itself, blogs on the same topic, and other blogs published by the same user. Then a unified neural network model is constructed with self-attention mechanism for Task 2. The model is based on multi-scale convolutional neural networks with the aim to capture both local and global information for user profiling. Finally, we adopt a stacking model for predicting user growth value. According to SMP CUP 2017's metrics, our model achieved scores of 0.563, 0.378 and 0.751 on the three tasks, respectively.

This paper is organized as follows. Section 2 introduces User Profiling Technology Evaluation Campaign in details. Section 3 describes the framework of our system. We present the evaluation results in Section 4. Finally, Section 5 concludes the paper.

2. EVALUATION OVERVIEW

2.1 Data Set

The data set used in SMP CUP 2017 is provided by CSDN, which is one of the largest information technology communities in China. The CSDN data set consists of all user generated content and the behavior data from 157,427 users during 2015, which can be further divided into three parts:

1). 1,000,000 pieces of user blogs, involving blog ID, blog title and the corresponding content;
2). Six types of user behavior data, including *posting*, *browsing*, *commenting*, *voting up*, *voting down* and *adding favorites*, and the corresponding date and time information;
3). Relationship between users, which refers to the records of *following* and *sending private messages*.

More details about the size and type of the CSDN data set are shown in Table 1.

[①] https://github.com/LuJunru/SMPCUP2017_ELP

Table 1. Statistics of the evaluation data set.

	Attribute	Content	Size	Format
Blogs		Users' blogs	1,000,000	D0802938/Title/Content
Behavior	Post	Record of posting blogs	1,000,000	U0024827/D0874760/2015-02-05 18:05:49.0
	Browse	Record of browsing blogs	3,536,444	U0143891/D0122539/20150919 09:48:07
	Comment	Record of commenting on blogs	182,273	U0075737/D0383611/2015-10-30 11:18:32.0
	Vote up	Record of clicking a "like" button	95,668	U0111639/D0627490/2015-02-21
	Vote down	Record of clicking a "dislike" button	9,326	U0019111/D0582423/2015-11-23
	Add favorites	Record of adding blogs to a user's favoriates list	10,4723	U0014911/D0552113/2015-06-07 07:05:05
Relationships	Follow	Record of following relationships	667,037	U0124114/U0020107
	Send private messages	Record of sending private messages	46, 572	U0079109/U0055181/2015-12-24

Table 2 illustrates an example from the given data set.

Table 2. Sample of CSDN data set.

Attribute	Data sample
User ID	U00296783
Blog ID	D00034623
Blog content	Title and content.
Keywords	Keyword1: TextRank; Keyword2: PageRank; Keyword3: Summary
Interest tags	Tag1: Big data; Tag2: Data mining; Tag3: Machine learning
Post	U00296783/D00034623/20160408 12:35:49
Browse	D09983742/20160410 08:30:40
Comment	D09983742/20160410 08:49:02
Vote up	D00234899/20160410 09:40:24
Vote down	D00098183/20160501 15:11:00
Send private messages	U00296783/U02748273/20160501 15:30:36
Add favorites	D00234899/20160410 09:40:44
Follow	U00296783/U02666623/20161119 10:30:44
Growth value	0.0367

2.2 Tasks

Task 1: To extract three keywords from each document that can well represent the topic or the main idea of the document.

Task 2: To generate three labels to describe a user's interests, where the labels are chosen from a given candidate set (42 in total).

Task 3: To predict each user's growth value of the next six months according to his/her behavior of the past year, including the texts, the relationships and the interactions with other users. The growth value needs to be scaled into [0, 1], where 0 presents user drop-out.

2.3 Metrics

To assess the system effectiveness in completing the above-mentioned tasks, the following evaluation metrics are designed for each individual task.

Score$_1$ is defined to calculate the overlapping ratio between the extracted keywords and the standard answers, which can be computed in Equation (1):

$$Score_1 = \frac{1}{N} \sum_{i=1}^{N} \frac{|K_i \cap K_i^*|}{|K_i|}, \tag{1}$$

where N is the size of the validation set or the test set, K_i is the extracted keywords set from document i, and K_i^* is the standard keywords of document i. Note that it is defined that $|K_i| = 3$ and $|K_i^*| = 5$.

Score$_2$ denotes the overlapping ratio of model tagging and answers, which can be expressed by Equation (2):

$$Score_2 = \frac{1}{N} \sum_{i=1}^{N} \frac{|T_i \cap T_i^*|}{|T_i|}, \tag{2}$$

where T_i is the automatically generated tag set of user i, and T_i^* is the standard tags of user i. It is also defined that $|T_i| = 3$ and $|T_i^*| = 3$.

Score$_3$ is calculated by relative error between the predicted growth value and the real growth value of users, which can be expressed by Equation (3):

$$Score_3 = 1 - \frac{1}{N} \sum_{i=1}^{N} \begin{cases} 0, & v_i = 0, v_i^* = 0 \\ |v_i - v_i^*| / max(v_i, v_i^*), & otherwise \end{cases}, \tag{3}$$

where v_i is the predicted growth value of user i, and v_i^* is the real growth value of user i.

The overall score can be computed by Equation (4):

$$Score_{all} = Score_1 + Score_2 + Score_3 \tag{4}$$

3. SYSTEM OVERVIEW

The overall architecture of UIR-SIST is described in Figure 1. UIR-SIST system is comprised of four modules:

1). Preprocessing module: To read all blogs of training set and test set. It performs word segmentation, part-of-speech (POS) tagging, named entity recognition and semantic role labeling;

2). Keyword extraction module: To extract three keywords to represent the main idea of a blog, which can be captured from three aspects to generate the candidate keywords set, including the blog content, other blogs published by the same user, and the blogs on the same topic, as shown in the green part;

3). User interests tagging module: To construct a neural network combined with user content embedding and keyword and user tag embedding for user interests tagging, as shown in the red part;

4). User growth value prediction module: To incorporate users' interaction information and the behavior features into a supervised learning model for growth value prediction, as shown in the blue part.
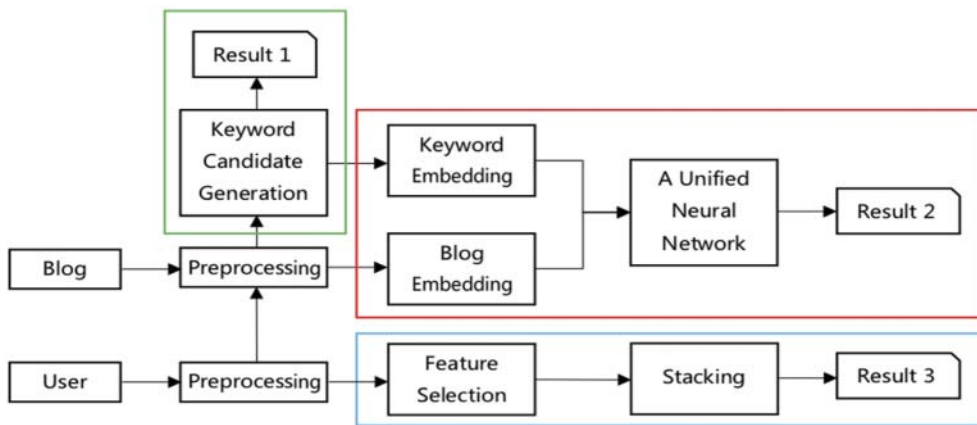
Figure 1. System architecture.

3.1 Keywords Extraction

The objective of Task 1 is to extract three keywords from each blog that can represent the main idea of the blog. In our opinion, the main idea can be extracted from the following three aspects, the blog itself, other blogs published by the same user, and the blogs on the same topic. Based on this assumption, we adopt three different models that can capture each aspect to generate a candidate keywords set, including *tf-idf*, TextRank and LDA, which are proved very effective in the relevant tasks. Then three keywords are extracted from the candidate set by using different rules.

We first adopt the classic *tf-idf* term weighting scheme to reflect the content of the blog itself. Then we rank the keywords based on the *tf-idf* score, and select the top 100 keywords to form the candidate keyword set.

Regarding the blogs on the same topic, we adopt TextRank approach [9] to cluster these blogs together. Meanwhile, all the keywords will be weighed during this process. We finally select the top 300 keywords.

Moreover, we utilize topic information to extract the keywords. Since 42 categories of tags are given in Task 2, we assume that these 42 topics are extracted from all the blogs. Therefore, we use Latent Dirichlet Allocation (LDA) model [10] to extract top 100 keywords for each category from 1,000,000 blogs, and thus obtain the interspecific distribution information of these 4,200 subject keywords.

In summary, we consider three aspects in order to reflect the blog content and obtain three independent candidate keywords sets, which are extracted through *tf-idf* model, TextRank model and LDA model. After that, we only save the intersection data set. In our training set of Task 1, about 5,000 keywords are provided, which are collected after extraction and deduplication.

A drawback of the classic *tf-idf* model is that it simply presupposes that the rarer a word is in corpus, the more important it is, and the greater its contribution is to the main idea of the text. However, when referring to a group of articles, which mainly use the same keywords and describe some similar concepts, the calculation results will have many errors. This is also the reason why we use *tf-idf* in the short blog, while we use the TextRank model in the long blog collection published by the same user.

In addition, in order to enhance its cross-topic analysis ability, we borrow the idea of 2016 Big Data & Computing Intelligence Contest sponsored by China Computer Federation (CCF)[②], and implement the improvements on the results of traditional *tf-idf* calculation, and obtain the result of *S-TFIDF*(*w*) by using Equation (5):

$$S - TFIDF(w) = TFIDF(w) * \left(\frac{1}{C_w} - \frac{1}{42} \right),$$ (5)

where C_w is the frequency of word w appearing in 42 categories.

3.2 User Interests Tagging

The objective of this task is to tag a user's interests with three labels from 42 given ones. We model this task with neural networks, and the model structure is shown in Figure 2. Each blog is represented by a blog embedding [11] through convolution and max-pooling layers. Then we obtain a user's content embedding from weighted sum of all of his or her blog embeddings. The weighted value of each blog embedding is counted by self-attention mechanism. Content embedding and keyword embedding are concatenated as user embedding, and finally fed to the output layer.

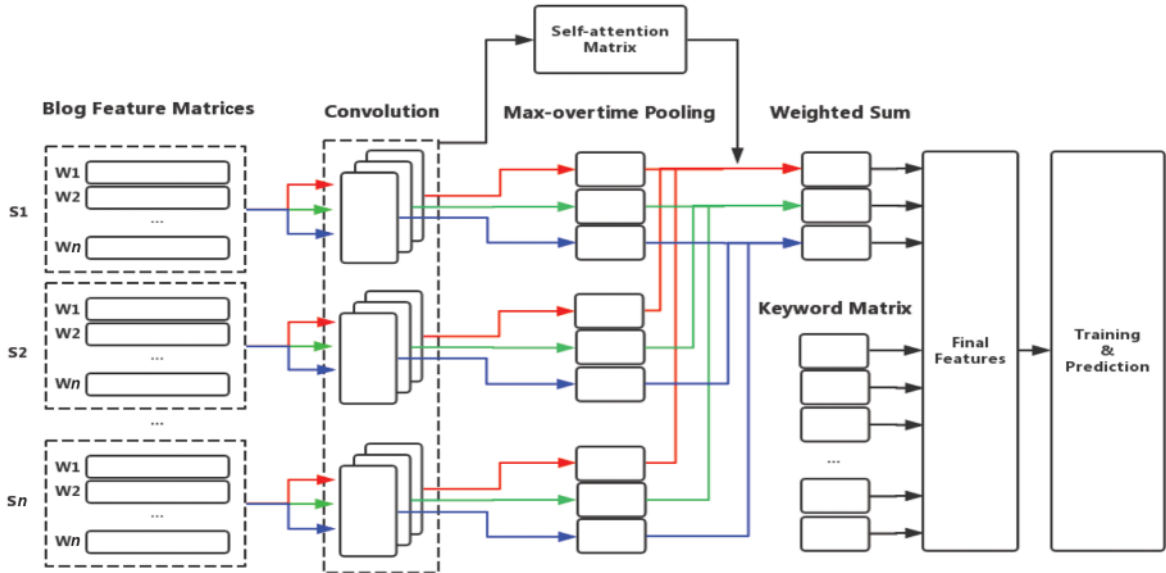[②] https://github.com/coderSkyChen/2016CCF_BDCI_Sougou

Figure 2. Framework of CNNs model based on weighted-blog-embeddings in Task 2.

In our system, a convolutional neural network (CNN) model is constructed for blog representation instead of a recurrent neural network (RNN), since more global information will be captured for indicating the user interests and the time efficiency will also be enhanced. It is widely acknowledged that a multi-scale convolutional neural network [12] has been implemented due to its outstanding achievement on computer vision [13], and TextCNNs designed by arraying word embedding vertically has also shown quite high effectiveness for natural language processing (NLP) tasks [14].

In our CNN model, we treat a blog as a sequence of words $x = [x_1, x_2, \ldots, x_1]$ where each one is represented by its word embedding vector, and returns a feature matrix S of the blog. The narrow convolution layer attached after the matrix is based on a kernel $W \in R^{kd}$ of width k, a nonlinear function f and a bias variable b as described by Equation (6):

$$h_i = f\left(W_{x_{i:j+k-1}} + b\right),\tag{6}$$

where $x_{i:j}$ refers specifically to the concatenation of the sequence of words' vectors from position i to position j. In this task, we use several kernel sizes to obtain multiple local contextual feature maps in the convolution layer, and then apply the max-overtime pooling [15] to extract some of the most important features.

The output of that is the low-dimensional dense and quantified representation of each single blog. After that, each user's relevant blogs are computable. We simply average their blogs' vectors to obtain the content embedding $c(u)$ for an individual user:

$$c(u) = \frac{1}{T}\sum_{i=1}^{T} s_i,$$ (7)

where T is the total number of a user's related blogs.

However, different sources of blogs imply the extent of a user's interest in different topics. For example, a blog posted by a user may be generated from an article written by himself, reposted by other users, or shared by users from another platform. It is natural that we may pay attention to these blogs in varying degrees when we infer this user's interests. Thus, a self-attention mechanism is introduced, which automatically assigns different weights to the value of each user's blog after training. The user context representation is given by weighted summation of all blogs' vectors:

$$\alpha = \frac{exp(e_i)}{\sum_{j=1}^{T} e_j},$$ (8)

$$e_i = v^T tanh(Ws_i + Uh_i),$$ (9)

$$c(u) = \sum_{i=1}^{T} \alpha_i h_i,$$ (10)

where α_i is the weight of the i-th blog, s_i is the one-hot source representation vector of the blog, $v \in R^n$, $W \in R^{n' \times m}$, $U \in R^{n' \times n}$, $s_i \in R^m$, $h_i \in R^n$, and m is the number of all source platforms.

When we finish a user's context representation, the keyword matrix of all blogs' keywords extracted by our model in Task 1 will be concatenated. The final features are the output of above whole feature engineering. Afterwards, an ANN layer trains the user embeddings from the training set and predicts probability distribution of users' interests among 42 tags in validation and test set according to their embeddings.

3.3 User Growth Value Prediction

According to the description of Task 3, the growth value can be estimated as the degree of activeness. Therefore, our basic idea is to incorporate a users' interaction information and his or her behavior statistical features into a supervised learning model. The procedure of Task 3 is demonstrated by Figure 3.
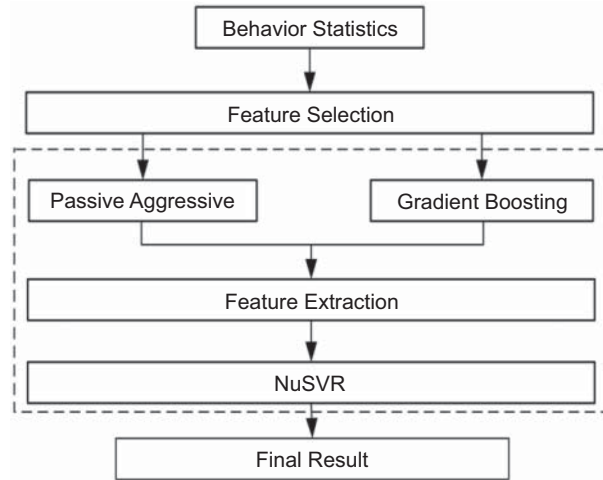
Figure 3. Framework of the stacking model in Task 3.

On the whole, we use a stacking framework [16] to enhance the accuracy of final prediction. After the basic behavior statistics analysis, the original features are selected as the inputs incorporated into the stacking model. Then, the stacking model is divided into two layers, the base layer and the stacking layer. In the base layer, we choose Passive Aggressive Regressor [17] and Gradient Boosting Regressor [18, 19] as the group of basic regressors due to their excellent performance. In the stacking layer, we still use the support vector machines (SVM) model, especially, the NuSVR model, which can control its error rate. Finally, we obtain the final results of user growth value.

3.3.1 Original Feature Selection

Figure 4 illustrates an example of the daily statistics of user behaviors, including posting, browsing, commenting, voting up, voting down, adding favorites, following, and sending private messages. To predict the user growth value, it is noted that the dynamic changes of behaviors along the time line are more useful. To avoid the sparse data problem, we adopt the monthly statistics of user behaviors rather than daily statistics.

	Post	Browse	Comment	Vote up	Vote down	Add	Follow	Send
U0002438	0	82	136	114	20	143	2	0
U0003009	2	280	8	10	0	0	3	10

Figure 4. Example of daily statistics of user behaviors. Note: "Add" refers to "add favoriates", and "send" refers to "send private messages".

Then we use correlation analysis to exclude the "vote down" behavior because of its negative contribution to model prediction. After that, through feature selection, we use the average, log calculation and growth rate of the original data to obtain features for the stacking model.

$$LOG(d) = log(d + 1), \tag{11}$$

$$GR(d_t) = \frac{d_{t+1} - d_t}{d_t + 1}, \tag{12}$$

where $LOG(d)$ represents the calculation results of data d after adjustment, and $GR(d_t)$ represents the calculation results of growth value from data d_t in month t to data d_{t+1} in month $t+1$.

3.3.2 PAR/GDR-NuSVR-Stacking Model (PGNS)

Once we have obtained monthly statistics and derivative features as described above, the combination of them will be sent as inputs into Passive Aggressive Regressor and Gradient Boosting Regressor independently. By averaging the predictions of those two base models, a new feature will be created and input into the stacking model NuSVR. Because of the inherent randomness of base models, we adopt a self-check mechanism of 10-fold cross validation.

If the trained model obtains a score higher than the threshold S^* under given scoring rules, we will enter the corresponding features of validation set or test set into the model for a prediction value, which will be saved into a *candidate set*. On the contrary, if the trained model obtains a 10-fold cross validation score that is lower than S^*, the model will be discarded and the program will return to the training session shown in the dotted box for a new round of training.

In order to reduce the errors of a single round of training, we set at least R^* rounds for training and add all predictions that obtain higher scores than S^* to the candidate set. According to our experience, the ratio of the size of a *candidate set* to R^* is about 0.45. When all rounds of trainings are completed, all predictions in the *candidate set* will be calculated to generate an average prediction as the final results.

4. EVALUATION

In our model, we first adopt Jieba® toolkit for Chinese word segmentation, and then train a word embedding with the dimensions of 300 [11].

Table 3 shows the comparison results of our proposed approach for Task 1. It is observed that the best results are achieved when data of all the three aspects are used for capturing the main ideas of blogs.

® https://github.com/fxsjy/jieba

Table 3. Comparison on Task 1 with different aspects.

Approach	Results
BI: Blog itself	0.505
ST: Same topic	0.371
SU: Same user	0.436
BI+ST+SU	**0.563**

Besides, we also test performance of our combined neural network with different embedding inputs. Note that to obtain the results of individual embedding, we train a new CNN model for blog embedding, and compute the similarity between blog content and keywords in the embedding representation. The experimental results are summarized in Table 4. It is observed that the embedding of blog content proves more effective than that of keywords, while they together achieve the best run.

Table 4. Comparison of different aspects on Task 2.

Approach	Results
Blog embedding	0.301
Keywords embedding	0.245
Blog + keywords embedding	**0.378**

Table 5 displays the overall performance of our system's best run on each individual task, which achieved the sixth place in the competition.

Table 5. Performance of UIR-SIST system in SMP CUP 2017.

	Task 1	Task 2	Task 3	Total
Training set (10 Fold)	0.610	0.390	0.765	1.765
Validation set	0.560	0.390	0.730	1.680
Test set	0.563	0.378	0.751	1.692

5. CONCLUSIONS AND FUTURE WORK

In this paper, we present our system built for the User Profiling Technology Evaluation Campaign of SMP CUP 2017. To complete Task 1, we propose to extract keywords from three aspects from a user's blogs, including the blog itself, blogs on the same topic, and other blogs published by the same user. Then a unified neural network model with self-attention mechanism is constructed for Task 2. The model is based on multi-scale convolutional neural networks with the aim to capture both local and global information for user profiles. Finally, we adopt a stacking model for predicting user growth value. According to SMP CUP 2017's metrics, our model runs achieved the final scores of 0.563, 0.378 and 0.751 on three tasks, respectively.

Future work includes analysis of the relationships between users and blogs. We only use the users' behavior in Task 2 in the current system, but the time when blogs are published is ignored. We plan to include network embedding into our model. Moreover, we will collect more blogs with real time information, and attempt to incorporate the time information into our weighting schema in those tasks.

AUTHOR CONTRIBUTIONS

B. Li (byli@uir.edu.cn) is the leader of the UIR-SIST system, who drew the whole framework of the system. J. Lu (lj1230@nyu.edu) was responsible for building the model for keyword extraction, while L. Chen (lec@boyabigdata.cn) and K. Meng (kmmeng@uir.edu.cn) were responsible for the model construction of user interests tagging. F. Wang (wangfengyi18@mails.ucas.ac.cn) summarized the user growth value prediction, while J. Xiang (xiang.j@husky.neu.edu) and N. Chen (nchen@uir.edu.cn) summarized the evaluation and made error analysis. X. Han (hanxu@cnu.edu.cn) drafted the whole paper. All authors revised and proofread the paper.

ACKNOWLEDGEMENTS

REFERENCES

[1] M. Ciot, M. Sonderegger, & D. Ruths. Gender inference of twitter users in non-English contexts. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, 2013, pp. 1136–1145. doi: 10.1126/science.328.5974.38.

[2] L. Wendy, & R. Derek. What's in a name? Using first names as features for gender inference in Twitter. In: Proceedings of the 2013 AAAI Spring Symposium: Analyzing Microtext, 2013, pp. 10–16. doi: 10.1103/PhysRevB.76.054113.

[3] W. Liu, F.A. Zamal, & D. Ruths. Using social media to infer gender composition of commuter populations. In: Proceedings of the International Conference on Weblogs and Social Media. Available at: http://www.ruthsresearch.org/static/publication_files/LiuZamalRuths_WCMCW.pdf.

[4] D. Rao, & D. Yarowsky. Detecting latent user properties in social media. In: Proceedings of the NIPS MLSN Workshop, 2010, pp. 1–7. doi: 10.1007/s10618-010-0210-x.

[5] M. Pennacchiotti, & A.M. Popescu. A machine learning approach to Twitter user classification. In: Proceedings of the Fifth International Conference on Weblogs and Social Media, 2011, 281–288. doi: 10.1145/2542214.2542215.

[6] M.D. Conover, J. Ratkiewicz, M. Francisco, B. Gonçalves, A. Flammini, & F. Menczer. Political polarization on Twitter. In: Proceedings of the Fifth International Conference on Weblogs and Social Media, 2011, 89–96. Available at: https://journalistsresource.org/wp-content/uploads/2014/10/2847-14211-1-PB.pdf?x12809.

[7] C. Tu, Z. Liu, & M. Sun. PRISM: Profession identification in social media with personal information and community structure. In: Proceedings of Social Media Processing, 2015, pp. 15–27. doi: 10.1007/978-981-10-0080-5_2.

[8] SMP CUP 2017. Available at: http://www.cips-smp.org/smp2017/.

[9] R. Mihalcea, & P. Tarau. TextRank: Bringing order into text. In: Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, 2004. Available at: http://www.aclweb.org/anthology/W04-3252.

[10] D.M. Blei, A.Y. Ng, & M.I. Jordan. Latent dirichlet allocation. Journal of Machine Learning Research 3(2003), 993–1022. Available at: https://dl.acm.org/citation.cfm?id=944937]&preflayout=flat#source.

[11] T. Mikolov, K. Chen, G. Corrado, & J. Dean. Efficient estimation of word representations in vector space. In: Proceedings of Workshop at International Conference on Learning Representations (LCLR), 2013, pp. 1–13. Available at: https://www.researchgate.net/publication/234131319_Efficient_Estimation_of_Word_Representations_in_Vector_Space.

[12] Y. LeCun, B. Boser, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard, & L.D. Jackel. Handwritten digit recognition with a backpropagation network. In: Proceedings of Advances in Neural Information Processing Systems, 1990, pp. 396–404. Available at: https://dl.acm.org/citation.cfm?id=109279%22.

[13] A. Krizhevsky, I. Sutskever, & G. Hinton. ImageNet classification with deep convolutional neural networks. In: Proceedings of Advances in Neural Information Processing Systems, 2012, pp. 1–9. doi: 10.1145/3065386.

[14] Y. Kim. Convolutional neural networks for sentence classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, 2014, pp. 1746–1751.

[15] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, & P. Kuksa. Natural language processing (almost) from scratch. The Journal of Machine Learning Research 12(2011), 2493–2537. doi: 10.1016/j.chemolab.2011.03.009.

[16] D.H. Wolpert. Original contribution: Stacked generalization. Neural Netw 5(2)(1992), 241–259. doi: 10.1016/S0893-6080(05)80023-1.

[17] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, & Y. Singer. Online passive-aggressive algorithms. Journal of Machine Learning Research 7(3)(2006), 551–585. Available at: http://www.jmlr.org/papers/v7/crammer06a.html.

[18] J.H. Friedman. Greedy function approximation: A gradient boosting machine. Annals of Statistics 29(5)(2001), 1189–1232. doi: 10.1214/aos/1013203451.

[19] J. Friedman. Stochastic gradient boosting. Computational Statistics & Data Analysis 38(4)(2002), 367–378. doi: /10.1016/S0167-9473(01)00065-2.

AUTHOR BIOGRAPHY

Junru Lu is currently a Master's degree candidate at the Center of Urban Science and Progress, New York University. He received his Bachelor's degree from University of International Relations in 2018. His research interests include natural language processing, text mining and social computing.
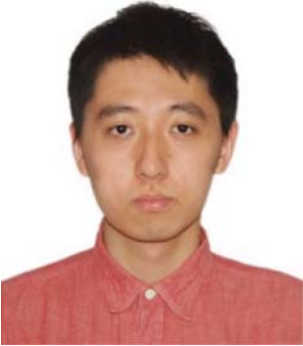
Le Chen received his Bachelor's degree from University of International Relations in 2018. He is now working as a data analyst in Beijing Boya Bigdata Co., Ltd. His research interests include text mining and social computing.

Kongming Meng is currently working as a data engineer in the DeepBrain Company. He received his Bachelor's degree from University of International Relations in 2018. His research interests include data mining and data analysis.

chinaXiv:202211.00460v1

Fengyi Wang is currently a graduate student at the University of Chinese Academy of Sciences (CAS). She received her Bachelor's degree from University of International Relations in 2018. Her research interests include natural language processing and social network analysis.

Jun Xiang is currently a graduate student in the program of Computer Systems Engineering, Northeastern University. She received her Bachelor's degree from University of International Relations in 2018. She has published two papers in international conferences and Chinese journals during her undergraduate studies.

Nuo Chen obtained her Bachelor's degree from the School of Information Science and Technology, University of International Relations in 2018. Her research interest is knowledge graph.

Xu Han is an Assistant Professor at Capital Normal University. She received her PhD degree in 2011. Her research interests are artificial intelligence and mobile cloud computing. She has published over 30 research papers in major international journals and conferences.
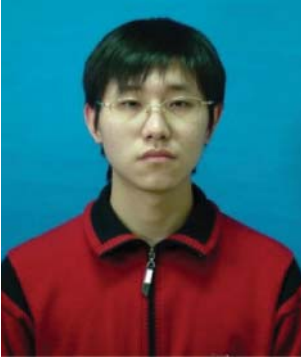
Binyang Li is an Associate Professor at School of Information Science and Technology, University of International Relations. He received his PhD degree from Chinese University of Hong Kong in 2012. His research interests include natural language processing, sentiment analysis and social computing. He has published over 50 research papers in major international journals and conferences.